```
                                          M 1

BBBBBBBBBBBB        AAAAAAAAA        SSSSSSSSSSSS  RRRRRRRRRRR     TTTTTTTTTTTTTT   LLL
BBBBBBBBBBBB        AAAAAAAAA        SSSSSSSSSSSS  RRRRRRRRRRR     TTTTTTTTTTTTTT   LLL
BBBBBBBBBBBB        AAAAAAAAA        SSSSSSSSSSSS  RRRRRRRRRRR     TTTTTTTTTTTTTT   LLL
BBB         BBB  AAA         AAA  SSS              RRR        RRR        TTT        LLL
BBB         BBB  AAA         AAA  SSS              RRR        RRR        TTT        LLL
BBB         BBB  AAA         AAA  SSS              RRR        RRR        TTT        LLL
BBB         BBB  AAA         AAA  SSS              RRR        RRR        TTT        LLL
BBB         BBB  AAA         AAA  SSS              RRR        RRR        TTT        LLL
BBB         BBB  AAA         AAA  SSS              RRR        RRR        TTT        LLL
BBBBBBBBBBBB     AAA         AAA     SSSSSSSSS     RRRRRRRRRRR           TTT        LLL
BBBBBBBBBBBB     AAA         AAA     SSSSSSSSS     RRRRRRRRRRR           TTT        LLL
BBBBBBBBBBBB     AAA         AAA     SSSSSSSSS     RRRRRRRRRRR           TTT        LLL
BBB         BBB  AAAAAAAAAAAAAAA              SSS  RRR  RRR              TTT        LLL
BBB         BBB  AAAAAAAAAAAAAAA              SSS  RRR  RRR              TTT        LLL
BBB         BBB  AAAAAAAAAAAAAAA              SSS  RRR  RRR              TTT        LLL
BBB         BBB  AAA         AAA              SSS  RRR     RRR           TTT        LLL
BBB         BBB  AAA         AAA              SSS  RRR     RRR           TTT        LLL
BBB         BBB  AAA         AAA              SSS  RRR     RRR           TTT        LLL
BBBBBBBBBBBB     AAA         AAA  SSSSSSSSSSSS     RRR        RRR        TTT        LLLLLLLLLLLLLL
BBBBBBBBBBBB     AAA         AAA  SSSSSSSSSSSS     RRR        RRR        TTT        LLLLLLLLLLLLLL
BBBBBBBBBBBB     AAA         AAA  SSSSSSSSSSSS     RRR        RRR        TTT        LLLLLLLLLLLLLL
```

```
**FILE**ID**BASFETCHD                                C 3




BBBBBBBB      AAAAAA      SSSSSSSS  FFFFFFFFFF  EEEEEEEEEE  TTTTTTTTTT    CCCCCCCC  HH      HH  DDDDDDDD
BBBBBBBB      AAAAAA      SSSSSSSS  FFFFFFFFFF  EEEEEEEEEE  TTTTTTTTTT    CCCCCCCC  HH      HH  DDDDDDDD
BB      BB  AA      AA  SS          FF          EE              TT      CC          HH      HH  DD      DD
BB      BB  AA      AA  SS          FF          EE              TT      CC          HH      HH  DD      DD
BB      BB  AA      AA  SS          FF          EE              TT      CC          HH      HH  DD      DD
BB      BB  AA      AA  SS          FF          EE              TT      CC          HH      HH  DD      DD
BBBBBBBB    AA      AA    SSSSSS    FFFFFFFF    EEEEEEEE        TT      CC          HHHHHHHHHH  DD      DD
BBBBBBBB    AA      AA    SSSSSS    FFFFFFFF    EEEEEEEE        TT      CC          HHHHHHHHHH  DD      DD
BB      BB  AAAAAAAAAA          SS  FF          EE              TT      CC          HH      HH  DD      DD
BB      BB  AAAAAAAAAA          SS  FF          EE              TT      CC          HH      HH  DD      DD
BB      BB  AA      AA          SS  FF          EE              TT      CC          HH      HH  DD      DD      ....
BB      BB  AA      AA          SS  FF          EE              TT      CC          HH      HH  DD      DD      ....
BBBBBBBB    AA      AA  SSSSSSSS    FF          EEEEEEEEEE      TT        CCCCCCCC  HH      HH  DDDDDDDD        ....
BBBBBBBB    AA      AA  SSSSSSSS    FF          EEEEEEEEEE      TT        CCCCCCCC  HH      HH  DDDDDDDD        ....

LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II        SSSSSS
LL              II        SSSSSS
LL              II              SS
LL              II              SS
LL              II              SS
LL              II              SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

```
0 MODULE BAS$FETCH_DESC (                         ! Fetch descriptor from array
0                 IDENT = '1-002'                 ! File: BASFETCHD.B32 Edit: PLL10002
0                 ) =
1 BEGIN
1 !
1 !****************************************************************************
1 !*                                                                          *
1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
1 !*  ALL RIGHTS RESERVED.                                                    *
1 !*                                                                          *
1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
1 !*  TRANSFERRED.                                                            *
1 !*                                                                          *
1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
1 !*  CORPORATION.                                                            *
1 !*                                                                          *
1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
1 !*                                                                          *
1 !*                                                                          *
1 !****************************************************************************
1 !
1
1 !++
1 ! FACILITY:  BASIC Language Support
1 !
1 ! ABSTRACT:
1 !
1 !       Fetch an element from an array of descriptors.  Return the
1 !       address of the descriptor.
1 !
1 ! ENVIRONMENT:  VAX-11 User Mode
1 !
1 ! AUTHOR: Pamela L. Levesque, CREATION DATE: 2-Mar-1982
1 !
1 ! MODIFIED BY:
1 !
1 ! 1-001 - Original.  PLL 2-Mar-1982
1 ! 1-002 - Offset for 1st index is 1, not 2.  PLL 19-Mar-1982
1 !--
1
1 !<BLF/PAGE>
```

```
1 !
1 ! SWITCHES:
1 !
1
1 SWITCHES ADDRESSING_MODE (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);
1
1 !
1 ! LINKAGES:
1 !
1 !     NONE
1 !
1 !
1 ! TABLE OF CONTENTS:
1 !
1
1 FORWARD ROUTINE
1     BAS$FETCH_DESC;                               ! Fetch descriptor from array
1
1 !
1 ! INCLUDE FILES:
1 !
1
1 REQUIRE 'RTLIN:RTLPSECT';                         ! Macros for defining psects
1
1 LIBRARY 'RTLSTARLE';                              ! System symbols
1
1 !
1 ! MACROS:
1 !
1 !       NONE
1 !
1 ! EQUATED SYMBOLS:
1 !
1 !       NONE
1 !
1 ! PSECTS:
1 !
1 DECLARE_PSECTS (BAS);                             ! Declare psects for BAS$ facility
1 !
1 ! OWN STORAGE:
1 !
1 !       NONE
1 !
1 ! EXTERNAL REFERENCES:
1 !
1 EXTERNAL ROUTINE
1     BAS$$STOP : NOVALUE;                          ! Signal fatal error
1
1 EXTERNAL LITERAL
1     BAS$K_ARGDONMAT : UNSIGNED (8),
1     BAS$K_NOTIMP : UNSIGNED (8),
1     BAS$K_SUBOUTRAN : UNSIGNED (8),
1     BAS$K_TOOFEWARG : UNSIGNED (8),
1     BAS$K_TOOMANARG : UNSIGNED (8);
1
1
```

```
1 GLOBAL ROUTINE BAS$FETCH_DESC (                       ! Fetch descriptor from array
1         DESCRIP,                                      ! The descriptor
1         INDEX1                                        ! First index
1     ) : =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       Given a descriptor for the array and the indices, calculate
1 !       the address of an element.  This element will be a descriptor.
1 !       Take into account that this may be a FORTRAN array.  This routine
1 !       does not handle virtual arrays.
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !       DESCRIP.rx.da   The descriptor of the array
1 !       INDEX1.rl.v     The first index into the array.  More indicies
1 !                       may follow this one in the calling sequence.
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !       NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !       NONE
1 !
1 ! ROUTINE VALUE:
1 !
1 !       The address of the descriptor is returned
1 !
1 ! COMPLETION CODES:
1 !
1 !       NONE
1 !
1 ! SIDE EFFECTS:
1 !
1 !       Signals if an error is encountered.
1 !
1 !--
1
2    BEGIN
2
2    BUILTIN
2         ACTUALCOUNT,
2         ACTUALPARAMETER;
2
2    LOCAL
2         INDEX_VALUE,
2         VALUE_LOCATION,
2         MULTIPLIERS : REF VECTOR,
2         BOUNDS : REF VECTOR,
2         LOW_INDEX,
2         HIGH_INDEX,
2         INDEX_INCR,
2         INDEX_NUMBER;
2
```

```
2       MAP
2           DESCRIP : REF BLOCK [8, BYTE];
2
2   !+
2   ! Be sure the number of array subscripts matches the number of
2   ! indicies given to us.
2   !-
2
3       IF ((ACTUALCOUNT () - 1) NEQU .DESCRIP [DSC$B_DIMCT])
2       THEN
3           BEGIN
3
4           IF ((ACTUALCOUNT () - 1) LSSU .DESCRIP [DSC$B_DIMCT])
3           THEN
3               BAS$$STOP (BAS$K_TOOFEWARG)
3           ELSE
3               BAS$$STOP (BAS$K_TOOMANARG);
3
2           END;
2
2   !+
2   ! The coefficients and bounds must be present.
2   !-
2
2       IF ( NOT (.DESCRIP [DSC$V_FL_COEFF] AND .DESCRIP [DSC$V_FL_BOUNDS])) THEN BAS$$STOP (BAS$K_ARGDONMAT);
2
2       MULTIPLIERS = DESCRIP [DSC$L_M1];
2       BOUNDS = DESCRIP [DSC$L_M1] + (%UPVAL*.DESCRIP [DSC$B_DIMCT]);
2   !+
2   ! Compute the lower and upper index numbers based on how the array
2   ! is stored.
2   !-
2
3       IF (.DESCRIP [DSC$V_FL_COLUMN])
2       THEN
3           BEGIN
3           LOW_INDEX = .DESCRIP [DSC$B_DIMCT];
3           HIGH_INDEX = 1;
3           INDEX_INCR = -1;
3           END
2       ELSE
3           BEGIN
3           LOW_INDEX = 1;
3           HIGH_INDEX = .DESCRIP [DSC$B_DIMCT];
3           INDEX_INCR = 1;
2           END;
2
2       INDEX_NUMBER = .LOW_INDEX - .INDEX_INCR;
2   !+
2   ! Compute the linear index from the indices provided.
2   !-
2       VALUE_LOCATION = 0;
2
2       WHILE ((INDEX_NUMBER = .INDEX_NUMBER + .INDEX_INCR) NEQ (.HIGH_INDEX + .INDEX_INCR)) DO
3           BEGIN
3           INDEX_VALUE = ACTUALPARAMETER (.INDEX_NUMBER + 1);
3
```

```
;  221          0313  5             IF ((.INDEX_VALUE LSS .BOUNDS [(.INDEX_NUMBER - 1)*2])  !
;  222          0314  4                 OR (.INDEX_VALUE GTR .BOUNDS [((.INDEX_NUMBER - 1)*2) + 1]))
;  223          0315  3             THEN
;  224          0316  3                 BAS$$STOP (BAS$K_SUBOUTRAN);
;  225          0317  3
;  226          0318  3             VALUE_LOCATION = (.VALUE_LOCATION*.MULTIPLIERS [.INDEX_NUMBER - 1]) + .INDEX_VALUE;
;  227          0319  2             END;
;  228          0320  2
;  229          0321  2         VALUE_LOCATION = (.VALUE_LOCATION*.DESCRIP [DSC$W_LENGTH]) + .DESCRIP [DSC$A_A0];
;  230          0322  2
;  231          0323  2         RETURN .VALUE_LOCATION;
;  232          0324  2
;  233          0325  1         END;                                            ! end of BAS$FETCH_DESC


                                                                .TITLE  BAS$FETCH_DESC
                                                                .IDENT  \1-002\

                                                                .EXTRN  BAS$$STOP, BAS$K_ARGDONMAT
                                                                .EXTRN  BAS$K_NOTIMP, BAS$K_SUBOUTRAN
                                                                .EXTRN  BAS$K_TOOFEWARG
                                                                .EXTRN  BAS$K_TOOMANARG

                                                                .PSECT  _BAS$CODE,NOWRT,  SHR,  PIC,2

                                      07FC 00000                .ENTRY  BAS$FETCH_DESC, Save R2,R3,R4,R5,R6,R7,R8,- ; 0199
                                                                        R9,R10                                      ;
                   5A 00000000G  00   9E 00002                  MOVAB   BAS$$STOP, R10                              ;
                   50            6C   9A 00009                  MOVZBL  (AP), R0                                    ; 0264
                                 50   D7 0000C                  DECL    R0                                          ;
                   55       04   AC   D0 0000E                  MOVL    DESCRIP, R5                                 ;
                   52       0B   A5   9A 00012                  MOVZBL  11(R5), R2                                  ;
                   52            50   D1 00016                  CMPL    R0, R2                                      ;
                                 17   13 00019                  BEQL    3$                                          ;
                   50            6C   9A 0001B                  MOVZBL  (AP), R0                                    ; 0268
                                 50   D7 0001E                  DECL    R0                                          ;
                   52            50   D1 00020                  CMPL    R0, R2                                      ;
                                 06   1E 00023                  BGEQU   1$                                          ;
                   7E       00G  8F   9A 00025                  MOVZBL  #BAS$K_TOOFEWARG, -(SP)                     ; 0270
                                 04   11 00029                  BRB     2$                                          ;
                   7E       00G  8F   9A 0002B 1$:              MOVZBL  #BAS$K_TOOMANARG, -(SP)                     ; 0272
                   6A            01   FB 0002F 2$:              CALLS   #1, BAS$$STOP                               ;
     05       0A   A5            06   E1 00032 3$:              BBC     #6, 10(R5), 4$                              ; 0280
                            0A   A5   95 00037                  TSTB    10(R5)                                      ;
                                 07   19 0003A                  BLSS    5$                                          ;
                   7E       00G  8F   9A 0003C 4$:              MOVZBL  #BAS$K_ARGDONMAT, -(SP)                     ;
                   6A            01   FB 00040                  CALLS   #1, BAS$$STOP                               ;
                   54       14   A5   9E 00043 5$:              MOVAB   20(R5), MULTIPLIERS                         ; 0282
                   56       14 A542   DE 00047                  MOVAL   20(R5)[R2], BOUNDS                          ; 0283
     0B       0A   A5            05   E1 0004C                  BBC     #5, 10(R5), 6$                              ; 0289
                   51            52   D0 00051                  MOVL    R2, LOW_INDEX                               ; 0292
                   50            01   D0 00054                  MOVL    #1, HIGH_INDEX                              ; 0293
                   57            01   CE 00057                  MNEGL   #1, INDEX_INCR                              ; 0294
                                 09   11 0005A                  BRB     7$                                          ; 0289
                   51            01   D0 0005C 6$:              MOVL    #1, LOW_INDEX                               ; 0298
                   50            52   D0 0005F                  MOVL    R2, HIGH_INDEX                              ; 0299
                   57            01   D0 00062                  MOVL    #1, INDEX_INCR                              ; 0300
```

```
              52         51           57  C3 00065 7$:      SUBL3   INDEX_INCR, LOW_INDEX, INDEX_NUMBER         ; 0303
                                      53  D4 00069          CLRL    VALUE_LOCATION                              ; 0307
              59         50           57  C1 0006B          ADDL3   INDEX_INCR, HIGH_INDEX, R9                  ; 0309
                         52           57  C0 0006F 8$:      ADDL2   INDEX_INCR, INDEX_NUMBER                    ;
                         59           52  D1 00072          CMPL    INDEX_NUMBER, R9                            ;
                                      2A  13 00075          BEQL    11$                                         ;
                         58      04 AC42  D0 00077          MOVL    4(AP)[INDEX_NUMBER], INDEX_VALUE            ; 0311
              50         52           01  78 0007C          ASHL    #1, INDEX_NUMBER, R0                        ; 0313
                    F8 A640           58  D1 00080          CMPL    INDEX_VALUE, -8(BOUNDS)[R0]                 ;
                                      07  19 00085          BLSS    9$                                          ;
                    FC A640           58  D1 00087          CMPL    INDEX_VALUE, -4(BOUNDS)[R0]                 ; 0314
                                      07  15 0008C          BLEQ    10$                                         ;
                         7E      00G  8F  9A 0008E 9$:      MOVZBL  #BAS$K_SUBOUTRAN, -(SP)                     ; 0316
                         6A           01  FB 00092          CALLS   #1, BAS$$STOP                               ;
              50         53      FC A442  C5 00095 10$:     MULL3   -4(MULTIPLIERS)[INDEX_NUMBER], -            ; 0318
                                                                    VALUE_LOCATION, R0                          ;
              53         50           58  C1 0009B          ADDL3   INDEX_VALUE, R0, VALUE_LOCATION             ;
                                      CE  11 0009F          BRB     8$                                          ; 0309
                         50           65  3C 000A1 11$:     MOVZWL  (R5), R0                                    ; 0321
                         50           53  C4 000A4          MULL2   VALUE_LOCATION, R0                          ;
              53         50      10   A5  C1 000A7          ADDL3   16(R5), R0, VALUE_LOCATION                  ;
                         50           53  D0 000AC          MOVL    VALUE_LOCATION, R0                          ; 0323
                                          04 000AF          RET                                                 ; 0325

; Routine Size:  176 bytes,    Routine Base:  _BAS$CODE + 0000


;  234          0326  1
;  235          0327  1 END                                            ! end of module BAS$FETCH_DESC
;  236          0328  1
;  237          0329  0 ELUDOM




;
;                                 PSECT SUMMARY
;
;     Name                      Bytes                     Attributes
;
;   _BAS$CODE                       176  NOVEC,NOWRT,  RD ,  EXE,  SHR,  LCL,  REL,  CON,  PIC,ALIGN(2)


;
;                          Library Statistics
;
;                                         -------- Symbols --------        Pages        Processing
;      File                               Total    Loaded   Percent        Mapped       Time
;
;   _$255$DUA28:[SYSLIB]STARLET.L32;1      9776        7         0          581          00:01.1
;                                    COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACE/LIS=LIS$:BASFETCHD/OBJ=OBJ$:BASFETCHD MSRC$:BASFETCHD/UPDATE=(ENH$:BASFETCHD
;       )

; Size:           176 code + 0 data bytes
; Run Time:          00:06.1
; Elapsed Time:      00:14.3
; Lines/CPU Min:      3257
; Lexemes/CPU-Min: 15405
; Memory Used:  84 pages
; Compilation Complete
```

0023 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

BASFREE
LIS

BASEXITHA
LIS

BASFETCHD
LIS

BASFORINT
LIS

BASGETRFA
LIS

BASFETCHA
LIS

BASGET
LIS

BASFSP
LIS

BASFIND
LIS

BASFORMAT
LIS

BASHANDLE
LIS